RES MILITARIS
REVUE EUROPEENNE D ETUDES EUROPEAN JOURNAL OF MILITARY STUDIES
Social Science Journal

Survey: Delivering Cloud Services based SLAs on Open Source Cloud

By

Zidje Parfait Guy Patrick
Research Scholar, Department of Computer Science & Engineering, Koneru Lakshmaiah Education Foundation, Andhra Pradesh, INDIA. 522502
Email: 183030052@kluniversity.in

Dr. KVV Satyanarayana
Professor, Department of Computer Science & Engineering, Koneru Lakshmaiah Education Foundation, Andhra Pradesh, INDIA. 522502
Email: kopparti@kluniversity.in

Dr. Bi Jean Baptiste Gouho
Assistant Professor, Department of Mathematics and Computer Sciences, Université Nangui Abrogoua, Higher Education, Abidjan, Côte D'ivoire
Email: gouhojb.sfa@univ-na.ci

Abstract

Cloud computing allows enterprises or users to access resource pools on demand service through the Internet network. In fact, this trend has been materialized as a low-cost way to develop IT infrastructure solutions. However copyrighted cloud service providers seem very costly due to the services required. However, Open-Source cloud systems, provide affordable, reliable, powerful and the cheapest solutions. For this purpose, our investigations will be focusing on Delivering Cloud Services based SLAs on Open-Source Cloud, built and implemented within open sources tools and accessible through internet. On the other hand, using Open-Source Cloud platforms for delivering cloud Services triggers some severe issues Cloud vendors and Cloud consumers. Then, an entity of solution named SLA which is used to solve most issues between those main cloud actors. To accost these issues, we suggest Service Level Agreements (SLAs) that offer best profits for both cloud actors, and we brought out investigations by classifying open sources cloud platforms. The results reveal the effectiveness of SLA for both cloud actors such as cloud vendor and cloud consumer.

Keywords: Cloud Computing, Open-source Cloud, Cloud Services, SLA, PaaS, IaaS, SaaS

Introduction

Nowadays, Cloud Computing is the dominant general resolve computing paradigm, and OpenStack, OpenStack, Eucalyptus, Nimbus, OpenNebula, Abicloud [7] are the most popular open source cloud for private and public clouds. Different stacks of Cloud executions such as Networking, storage and application are considered as ones of the innovative research areas of the Cloud Computing research community. The main challenges in a cloud environment concern not only ensuring quality of service (QoS) and security, but also managing them in accordance with the corresponding service level agreements (SLAs). Cloud computing, that provides low-priced computing resources on the pay-per use base, is punctually succeeding dynamism as a substitute for old-trended IT Infrastructure-based enterprises (On-Premise). As increasingly enterprises wander their IT infrastructure towards Cloud computing trend, SLAs between cloud customers and Cloud vendors appear as important component to reckon on.

Owing to the dynamicity of the Cloud computing, endless supervising of Quality of Service (QoS) aspects is required to distinguish SLAs [1]. Cloud computing technology is a distributed computing environment that provides the cloud as an on-demand service to customers. The rapid evolution of cloud computing is due to its service delivery methods which offer huge benefits to cloud customers. Open-source software and an excellent operating tool that interacts like a large cloud computing operating system which has been automatically implemented to manage a large number of computing devices and which is capable of administering huge software and hardware resources of computing power, storage technologies and networking in an environment such as a virtualized data center by example OpenStack is one of powerful open-sources cloud framework for handling private and public clouds for computing, storage and network resources. Based on the user survey reports[2]. Most independent cloud service providers use OpenStack as open source due to the power of its core deployment service modules, as shown in the following figure:

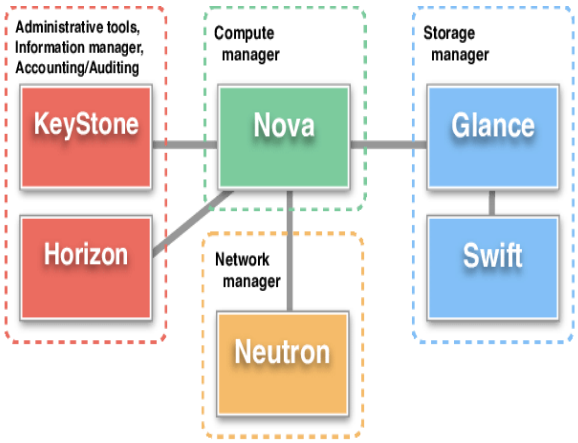

Figure 1. *OpenStack Modules*

In fact, OpenStack is a cloud computing implementation tool that offers software and hardware resources as an IaaS cloud infrastructure provider. Its main modules above are Horizon, Keystone, Nova, Glance, Swift, Neutron, Cinder, Heat and Ceilometer, were implemented in order to carry out all the deployment procedures while first respecting the NIST standards in order to meet expectations of end-users. OpenStack is open-source framework or operating system cloud. It is fundamentally an global association of cloud computing engineers and developers. Its goal is to implement several solutions for various kinds of cloud that are simple to implement, extremely accessible and wealthy in features. It offers Infrastructure as service (IaaS) solutions. Via OpenStack the cloud infrastructure is implemented with an architecture along 3 main nodes such as Controller, Compute and Network. Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) have the capabilities to supply all service execution stacks to the cloud consumer.

SaaS: with all the execution stacks while using the cloud vendor’s applications, executing on the provider's implemented cloud infrastructure, the end-user has no control over their core cloud infrastructure resources containing network layers, server layers, operating layers, storage layers or even specific application capabilities, except can be application-specific application structure parameters restricted [9].

PaaS: Applications of this type of cloud are deployed and implemented by consumers based on the sustained programming languages integrated by cloud vendors themselves. The

existing set of APIs bounds the capabilities to set the platform. Some of its examples are Google app engine [10], Rackspace [11], Force.com [12], and Microsoft Azure Platform [13].

IaaS: In this service of cloud policy, the cloud user consumes a virtual resources integrated to supply computing resources such as storage, network, and processing units, so that the deployment environment for their software systems can be settled. In this system, consumers are provided the flexibility to handle and control the pool of resources needed, which comprise applications, operating systems, and middleware. Some examples like OpenNebula [15], Openstack [16], and Eucalyptus [16].

On the other hand, the service level agreement (SLA) has the capabilities to describe the assurance, Quality of services (QOS), availability and scalability of cloud services in the cloud end-users and the cloud vendors based on the resources provisioning. Hardware and software and services supplied to end-users. Due to the requests for speedily growing services in the cloud, the demand for resources keep increasing constantly. Resource planning methods offer a greater allocation of resources, which often results in excessive resources creating SLA violations [4]. To this end, an architecture has been developed to provide cloud services in good performance to allow better use of allocated resources, to decrease the rate of SLA violations in order to multiply the income of cloud users and providers. Our approach is intended as follows: Section 2 explains some preliminary definitions of some cloud entities. Section 3 develops the allocation and planning parameters for SLAs, classifying and comparing most popular and used open-sources cloud and section 5 will help to understand the work and presents the future work.

Preliminary Definitions

Cloud Computing

Cloud computing is a basic computing model that provides intercommunication between divers shared platforms. Cloud computing operates the perception of virtualization technology. Concept Virtualization improves the flexibility, elasticity, capacity and scalability of organization's cloud IT infrastructure, which enhances the service availability for cloud users [8]. Cloud users can simply use the resources of their infrastructure without having to physically or materially install them. Cloud computing is a shared platform that supplies a dynamic resource on demand-service to customers. Virtualized cloud computing technology has become a very important step and deals with many services on the Internet. Its main objectives are high performance, scalability, fault tolerance, high availability, reliability and ease of use, monitoring, management and the provision of efficiency and economy. The hardware is compact for service, fully incorporated, and each software execution stack on the hardware is provided as entity of performance measurement and quality of service. Let us refer to the below architecture:

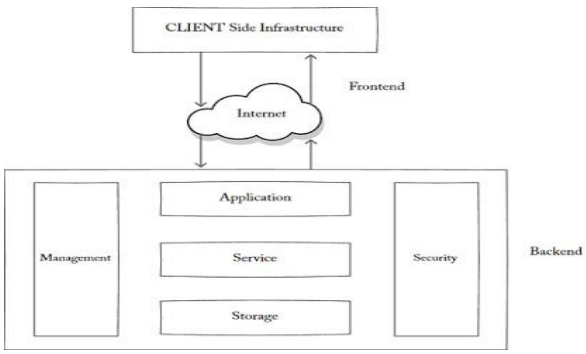

Figure 2. *Cloud Computing Architecture*

The fast growth of cloud trend is owing to its policy nature which brings numerous profits to enterprises. It has been a hot subject in the area of investigation and sciences since numerous years. Cloud computing has been a very fascinating and striking research policy in recent years. Most sector organizations are migrating towards this trend that permits end-users to access a pool of mutual IT resources such as storage, networks, applications and services might be rapidly supplied [5]. It exists some vital functionalities of cloud:

- Wide network access
- Self-service
- Self- Healing Management
- Pooling of resources
- Measured service
- Rapid elasticity, etc.

Mostly, four disposition models of cloud services are:

- Private cloud
- Public cloud
- Community cloud
- Hybrid cloud

Cloud computing mainly includes three service delivery models which are:

- Infrastructure as a service (IaaS)
- Software as a service (SaaS)
- Platform as a service (PaaS) [2], [3]

Mentioned in the following figure:

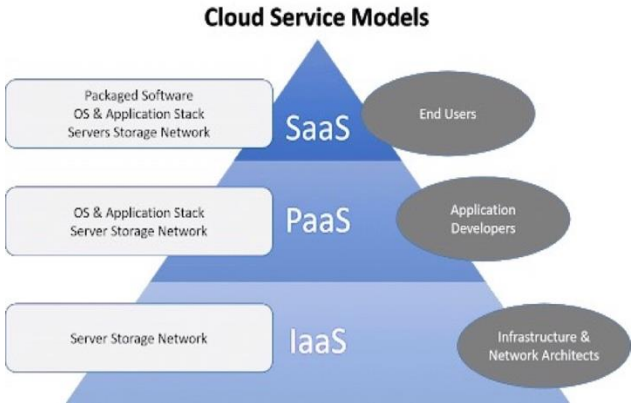

Figure 3. *Cloud Service Models*

The growing utilization of cloud computing renders the implementations of high-quality and high-performance cloud-based applications a serious area of research. Cloud supplies low-priced IT resources as they go, is rapidly expanding to replace traditional IT infrastructure organizations that we named On-Premise IT infrastructure. As increasingly utilization, users move their applications to cloud field, SLAs between cloud clients and cloud vendors are being an important component to reckon. Owing to the dynamic environment of the Cloud, boundless supervising of Quality-of-Service characteristics is needed to distinguish SLAs.

SLA Service Level Agreement

Cloud technology needs a perfect established SLA engaged by the cloud service customer and dedicated by service vendor as a useful services declared prior. Approaching with a contract of service is unique of the most prominent steps of the cloud delivery processes. It is only an IT service contract. The meekest meaning is that it is a agreement conveyed and approved between the cloud provider who supplies the cloud service and the cloud end-user who uses the services supplied. Providing resources as an on-demand service in a scalable manner is a fundamental distinguishing feature of the cloud initiative. The delivery of service components within the cloud focuses on SLAs representing a signed contract between the cloud client and the service provider designating the standards of the contract, including the non-operational obligations of the services mentioned as quality of service, the obligations and sanctions for breach of the agreement. The following figure shows the structure of the cloud SLA:

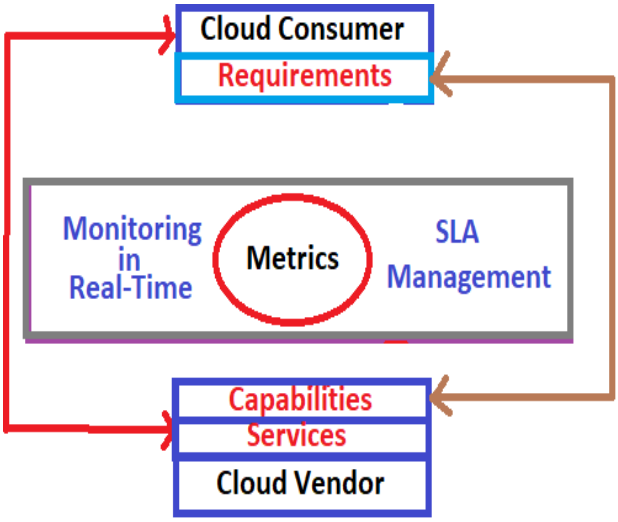

Figure 7. *Cloud SLA Service Level Agreement*

OpenStack Cloud Environnent

Today, Cloud technologies are the leading general IT paradigm and OpenStack [8] is one of best of Open-source cloud platforms. OpenStack is a framework open source cloud operating system that monitors huge pools of IT infrastructures in a data center whole everything is administered and supplied via APIs along with mutual validation of validation processes [7]. OpenStack [4] is a basic software for implementing a private or public cloud that supplies 3 types of service deliveries: Infrastructure as a Service (IaaS), Software as a Service (SaaS) and Platform as a Service (PaaS) model. It entails the main resources such as storage, networking, and processing in a data center in which the administrators administer through APIs via a dashboard. It exists plenty utilities in OpenStack, each providing diverse services accordingly. OpenStack is a powerful platform management that developers and researchers can utilize to implement and execute various kinds of clouds. OpenStack, by environment, is primarily used to implement a private cloud so that to except security criteria. It offers powerful virtual or logic servers and the services needed for cloud. It is primarily deployed as IaaS, which tends to provide workshops and hardware constituents for processing, storage and networking resources in a datacenter. It might also be compromised as cloud operating system that utilizes physical resources to virtualize them and lake them to make and manage services, public and private. By default, OpenStack offers some cloud-related functionalities such as networking, storage, image services, identity, etc., and are mutualized along with some other utilities to get custom cloud optimization to support built-in cloud applications. Compute is the

core the cloud producing and handling huge collections of logical servers and Object Storage is software for making scalable object storage by utilizing Cluster of service servers to store TBs or even PBs of data. It encompasses key components which are [11]:

- OpenStack framework is wholly an opened source code and hence can be efficient or adapted as per necessities. There are some corroboration procedures that shall be processed for the implementation and growth of new principles.
- Compatible and flexible: OpenStack is very flexible and more accessible and supports most virtualization solutions of the marketplace such as:
  - ESX
  - HYPER-V
  - KVM
  - QEMU
  - XENSERVER etc.

Scalability: OpenStack is very much accessible and scalable. it is already installed in company in which data volume is restrained in petabytes and scalable up to 1million physical machines Here are some reasons that response our requests:

- OpenStack is most importantly set of cloud tools that provides whole control over the cloud computation. Most of the platforms available in the market that aids in virtualization and cloud computation are all expensive.
- OpenStack might be deployed free of charge and might be personalized with the principle services to response the requirements. Therefore various companies possess their own OpenStack version.
- Besides, it has the capacities to resize to any level suitable.
- OpenStack Application Program Interface is built dynamic and more accessible and powerful with enhancements and perfections funded by several developers through the world. This also confirms good public support.
- Various great enterprises in the IT world like Huawei, Intel, Red Hat have adopted OpenStack on their cloud system management environment [6]:

The significant nomenclature which are frequently utilized in the rest of the module are debated shortly. These are prominent tools of OpenStack. However OpenStack possesses various parts. Moreover, due to its open-source nature plenty developers are able to fund to upgrade new modules for customized application. On the other hand to explain, the OpenStack opens around 9 key modules mentioned on fig 4:

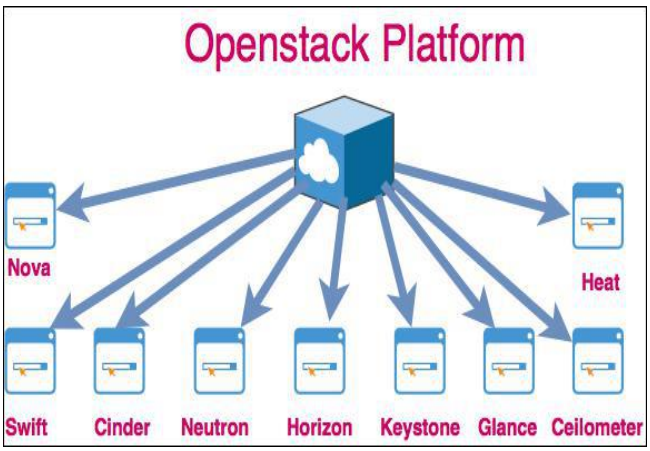

Figure 4. *9 Main Components of OpenStack*

- Nova: This is the central of openstack cloud. It handles a huge set of VMs and instances, which manages computing workload.
- Swift: is the OpenStack storage environment. It has been designed to schedule and store the set of objects and files. As an alternative of mentioning to the file and objects through the path, developers can as a substitute mention to them through a single identifier. These topics to a file or part of data and thus permit the OpenStack to handle where to store the files. It shrinks the exertion of the developers to figure out and concern about storage sharing.
- Cinder: or the block storage modules of OpenStack. These components in a method, referred to the former methods of localizing and accessibility particular localities on a disk.
- Neutron: is the module which activates networking system within OpenStack cloud environment. It safeguards and guarantees that every module in OpenStack is completely connected with other modules, to accomplish best connection between them.
- Horizon: is the set of web interfaces of OpenStack management. It supplies all the promises, potentials and functionalities for the administrators to admit and handle the cloud tools. It is the initial and primary module which everybody perceives upon preliminary to use OpenStack. Administrators must be able to manage all the modules over the API also, while Horizon is the main locus through which system administrators will interact to the OpenStack design.
- Keystone: is the constituent which offers the character services to all the administrators. It essentially encompasses an essential number of all the admin of the cloud environment, planned to the available and manageable of services. It offers a method for numerous admittances by consenting the developers to plan their current operator access procedures.
- Glance: supplies the image services in OpenStack cloud environment, wherever images mention to the virtual duplicates of the hard disks. It supports in trade with these images to be used as patterns although allocating new VM instances.
- Ceilometer: Ceilometer offers telemetry services to its cloud end-users. It achieves a close or local and measurement parameters as of every user of cloud services usage and offers an invoice for the services consumed. Ceilometer is a module to measure the utilization and statement the same to different operators.
- Heat: is a set of utilities which permits developers to store the cloud application necessities in a file so that all the essential resources for a program are accessible at hand. It thus offers an infrastructure to handle and deal the application. It is defined as orchestration mechanism within OpenStack [10].

Cloud architecture primarily emphases on a mutual characteristic of resources which are simulated and exposed to a cloud client on-demand services base. These cloud computing utilities are linked to disclose these resources over the Internet. However OpenStack cloud is used to figure private cloud and set out 3 service prototypes. Compensations of consuming OpenStack cloud platform:

- Abridged cost, low-priced
- Quicker distribution
- Quicker utilization
- Quicker Personalization and scalability.
- Elastic execution

SaaS Software as a Service cloud

SaaS, Software as a Service, are collectively a continuous cloud-based software completion administered by successful distributions. SaaS completions are common with a perfect pay-as-you-go pricing or pay-per use contribution and subscription. In terms of character, capacity and accessibility of evolution development SaaS is practical when you want software solutions for important but not critical business goals [9]:

- Cost: Internal-cost for developing a business utility tool is pretty high. Apart from the cost of development, there will be bugs & lot of duplications previously the solution can be installed. Most SaaS cloud solution vendors also take charge of hosting, which an extra cost is saved.
- Quick Deployment & removal: SaaS is quick & easy to deploy. Open-source requires some work before they can be deployed.
- Experimentation: SaaS is ideal for experimentation. Deploying open-source require a lot of early investment before deployment, which cannot be recovered in-case you decide to discontinue.

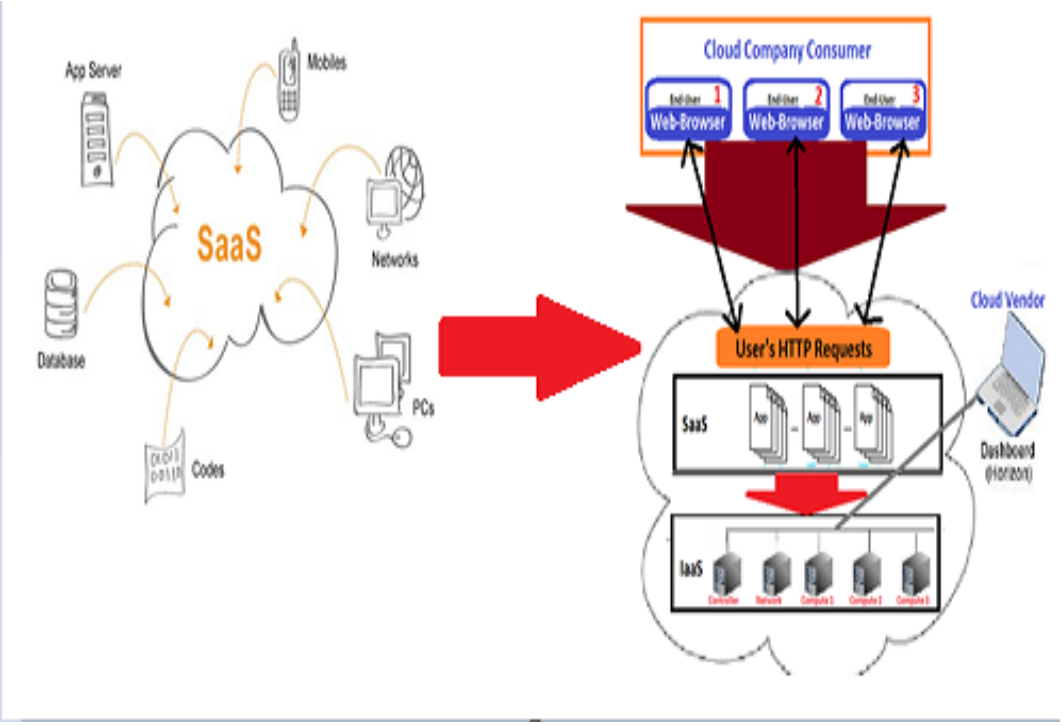

Figure 5. *SaaS Cloud Architecture based on OpenStack*

OpenStack Deployment Processes

We will explain the phases deploying private cloud by utilizing Packstack utilities [15]. Packstack is a framework that utilizes components to install and organize various stacks and portions of OpenStack on several pre-installed servers over SSH mechanically. Now only CentOS and Red Hat (RHEL) and compatible products of both are maintained. If you are nonexistence of resources and still need to know cloud basically, then Packstack is the top choice opened and existing. Packstack can be mounted on your laptop and you can effortlessly practice cloud environment within some moments. Packstack provides an easy way to deploy an OpenStack Platform environemnt on one or several machines because it is customizable through a answer file, which contains a set of parameters that allows custom configuration of underlying Openstack platform service. Now I will be arguing all the phases essential and mandatory to deploy Packstack on your local machine concerning these necessities on following the table 1 [17] [23]:

Table I: *Cluster Configuration Parameters On Openstack Cloud*

Physical Server Properties	Values
Number Of Nodes	4 (1 Master and 3 slaves)
Instances Of Openstack	Linux CentOS-7 server, 16 GB RAM and 200 GB Disk
Openstack Host Machine	Intel Xeon CPU X5670, 24 cores, 24GB RAM, Windows Server 2016, 64bit, 500 GB hardDisk
FQDN	Openstack.example
Hostname	Openstack

We are going to use Oracle VirtualBox (the latest version) and CentOS 7, which can be downloaded from:

http://centos.mirrors.estointernet.in/8.1.1911/isos/x86_64/CentOS-8.1.1911-x86_64-dvd1.iso

Packstack supplies a response as file template which shall be deployed into ALL IN ONE environment. You have to assign your parameter options through a textfile, mentioned as a response file.

Step 1: the physical host server

Firstly you need a base server on which you will create your entire Openstack cloud for which I have bought my server with RHEL 7.4

Step 2: Configure DNS Server

A DNS server is suggested prior you deploy OpenStack on CentOS 7 via Packstack.

Step 3: Controller VM Settings and Install Packstack

The Controller node is wherever utmost of the mutual OpenStack services and other tools execute. It provides Application Programming Interfaces, planning, and other mutual services for the cloud.

Step 4: Accessing OpenStack Dashboard

```
Preparing Swift proxy entries                        [ DONE ]
Preparing Swift storage entries                      [ DONE ]
Preparing Heat entries                               [ DONE ]
Preparing Heat CloudFormation API entries            [ DONE ]
Preparing Gnocchi entries                            [ DONE ]
Preparing Redis entries                              [ DONE ]
Preparing Ceilometer entries                         [ DONE ]
Preparing Aodh entries                               [ DONE ]
Adding Magnum manifest entries                       [ DONE ]
Preparing Puppet manifests                           [ DONE ]
Copying Puppet modules and manifests                 [ DONE ]
Applying 192.168.10.10_controller.pp
192.168.10.10_controller.pp:                         [ DONE ]
Applying 192.168.10.10_network.pp
192.168.10.10_network.pp:                            [ DONE ]
Applying 192.168.10.10_compute.pp
192.168.10.10_compute.pp:                            [ DONE ]
Applying Puppet manifests                            [ DONE ]
Finalizing                                           [ DONE ]

 **** Installation completed successfully ******
```

Figure 6. *OpenStack Dashboard of administration*

Right now you can install whatever utilities we need to utilize on the cloud and then can attach the cloud resources to other devises connected on the network. You can involvement services of cloud as software as a service by mounting application on cloud instances and platform as a service also. But this PackStack is just for trying out OpenStack because every time you must execute ./stack.sh command which means all services will start again.

SaaS as an Open Source Cloud

SaaS supplies applications already installed and deployed on a supplier's hardware and software resources. Open source SaaS cloud systems permit quick personalization, optimization and growth of the applications offered. There are various prominent SaaS cloud providing. Open source solutions are utilities in which source code are available free of charge [9]. These are really useful whenever an organization is searching for a gratis resolution and has a crew and a set of SaaS equipment offering applications already installed and deployed on a vendor's software and hardware resources. Those are really useful however an enterprise looks for a gratuitous utilities developers is released to personalize utilities to suit their needs. Good examples of open source software are Linux, MySql, Joomla, phpBB, myBB etc. Overall, open source software can be classified into two categories: development tools and commercial tools. Development tools, such as C ++, Java, MySql, php etc., are the cornerstone of software development and are generally supported by a non-profit research institute or a group of enthusiasts [9]. The other category, commercial tools, meets various commercial needs such as email, CRM, defect management, content management, human resource management, etc.

- Flexibility: Developers really open source solution because of their capability to customize the underlying code. Open source solutions are pretty good at fulfilling developer’s needs.
- Quality-code: Using open source methodology produces better-software, if there is a bug in the code, it can be quickly identified by the community.
- Reduce business risk: Business can reap-cost benefits. By using open source, they can avoid lock-in & heavy investment required for buying software. Open source solutions are also easier to customize to fulfill business needs.

Problems to Consider For Saas Providers’ Selection

In general your SaaS offering shall be aware about scalability and be able to make use of auto-scaling, which means your SaaS enabled application must be designed and implemented to be cloud ready. Cloud ready means the ability to scale. OpenStack provides HEAT for Auto-Scaling, but that's not enough, your SaaS offering must be aware to use auto-scaling. Enterprises and end-users who opt to utilize SaaS cloud services must take into account

various problems prior depending on SaaS provisioning. Those gaps are service functionality subjects, assimilation with on-premises services methods, SLA cooperation and agreement supervising, application change management, data access, and data security:

Service functionality: the enterprises must reassure the SaaS providing owns exactly and wholly all the functionalities and optimized utilities of the on-premise, inside hosted service in the circumstance of an on-premises service and all the needed functionalities in the circumstance of new services which will add along with services. Besides, the enterprise shall control either the SaaS offer is around a distinguished service with. SLA compromise and accordance supervising: the organization must to reassure that an SLA controls the service provisioning; the two actors must deal the standards of the SLA; it owns the needed evidences on the SLA decisions accessible from the SaaS; SLAs, or Service Level Agreements, are standards within the IT subcontracting area. The knowledge was to "guarantee" the selected IT service vendor would offer the promised performance levels or face disadvantages as fines (as credits). The guarantee let the organizations to converge its responsiveness and in-house resources in other value-add areas. But IT service vendors can reflect SLAs as something they simply must make and comprise in each main service agreements to make sure their end-users. In certain circumstances, they cannot plentifully provide SLAs understood at all, using basic language that doesn't work with the end-users excepted from performance requirements. That's why SLAs are frequently worth a bit more than the document they're printed on, particularly if there is no supplementary expressive fine structure.

Nowadays, with more enterprises embracing Software-as-a-Service (SaaS) and cloud computing models overall, IT managers are mainly excited to accomplish perfect standards for performance related to those put in place with their IT service vendors. SLAs can play an useful mental role in decreasing the worry certain organizations can feel around approving solutions that exist in the Cloud. Evidently, expressive SLA structures must be a fragment of all principal subscription and contribution agreements with all selected SaaS suppiers. This comprises all cloud providers. To do some your SaaS SLAs are evocative to your requirements, make sure you took into account:

- SLA meaningful
- UpTime
- DownTime
- Violation Detections
- Penalties
- Scalability
- Reporting

Key Components of Saas Cloud Openstack

Key Components of SaaS

So as to describe a perfect quality for the assessment of the SaaS cloud service, we need distinguish the key features of the SaaS service as follows.

Quality of Service (QoS)

Quality of service is a set of evaluation techniques linked to the level of services available and the service abilities of the supplier. The supplier's level of service includes:

- Accessibility

- business permanence
- performance
- usability
- consistency
- Trustworthiness

The vendor's service abilities include:

- data security capabilities
- law enforcement capabilities
- SaaS service management capabilities

Scalability

A multi-tenant infrastructure in SaaS facilitates rapid ramp-up depending on the load, unless without breaking SLAs.

Pay-Per-Use

It destines that users only pay for what they use such as: application and storage's resources, which allows them to have an efficient total cost. Pay-Per-Use is the main function of the SaaS cloud services.

2.Universal design of the SaaS service

The common design of the SaaS service is clarified in fig.9 [20]. Three characters of the SaaS service involves three roles explicitly:

- the service consumers (Cloud End-Users)
- the SaaS platform
- the application vendors

The service customer utilizes services via Internet, the SaaS platform manages the cloud set of resources and the SaaS services, and the software developer implement and deploys the services in the Cloud platform.

a)Cloud Consumers

Cloud Consumers is set of organization’s end-users that currently consumes the cloud services. Characteristically cloud clients use cloud services along with SLAs and pacts which are compromised between the cloud users and SaaS providers. In real-time, cloud consumers must frequently assess SaaS platform to make sure its service encounter the SLA requirements.

SaaS platform

SaaS platform offers the service to the consumer; its design comprises hardware and software infrastructure resources. Figure 9, In fact, the SaaS platform does not need to possess the physical resources; cloud consumers concern SaaS standards, must utilize other cloud vendors’ services, Infrastructure as a Service or Platform as a Service. The application design involves the concepts in the shared application architecture but further encompasses it to ease and to enable key functionalities of SaaS cloud services. The lowermost level of Application design is the data tier, above that is the application tier. In application tier, software service offers essential utilities; cloud consumer utilizes the pattern service to personalize their User-Interfaces, workflow. Those other modules is runtime mean which performs a prominent role in engendering renter particular service, the submodule of runtime comprises business logic engine, User-Interfaces engine, data engine and load-balancer. Vital to SaaS platform is the

management level and security and data privacy level. In SaaS cloud model, user's data is stored in the platform's datacenter, with the data of other cloud consumers, then platform shall make other labor to make sure data security and data privacy. Platform management implies some report regarding: Pay-As-You Pricing as invoicing, metering reports, capacity planning report, SLA management report, supervising report.

Application developer

The application developer implements, deploys, maintains and supervises the application services provided.

Advantages of SaaS over IaaS

IaaS, Infrastructure as a Service is managed by a platform for managing all material resources, in which SaaS services are accomplished at lower cost, install them quicker and reduce the cost of proprietorship related to constructing on your own infrastructure. This may aid you to implement, install and administer your SaaS cloud applications by empowering you to obtain new instances up and executing in minutes and reducing the effort involved in maintaining and updating them [14]. As the number of service clients is looked forward to develop in the future, it is dynamic for cloud service vendors to overwhelm this condition by being capable to distribute resources on an as-needed basis. So, they must get the means to supervise utilization of resources and assess several Quality of service metrics in order to be capable to satisfy cloud consumer's SLA requirements. Given the set of solution integration limits for SLA-based SaaS option and provision via APIs offered by each SaaS vendor, we determined to develop an environment that:

- Permit service customers to precise their functional and nonfunctional requests.
- Hide the heterogeneity of SaaS vendor's APIs from service customers.
- Permit executing QoS-driven choice of SaaS vendors.
- Permit applying SLA compromise on behalf of the service customers.
- Permit observing, supervising and evaluation of SLAs performance.

To assess the existing QoS for each type of cloud service metric they provide, service vendors must utilize supervising strategies to collect measurement data at designated statements. By accumulating gathered data, the SaaS vendors may define the assessment of each QoS points and controls its accordance along with the SLA. In the event of a significant drop in the actual QoS offer, the SaaS vendor is in charge of making suitable decisions such as adding additional resources to prevent any lament from the cloud clients. Figure 4 shows a specific architecture of a SaaS vendor, which designs several management functions which cooperate to offer characterized services to cloud customers.

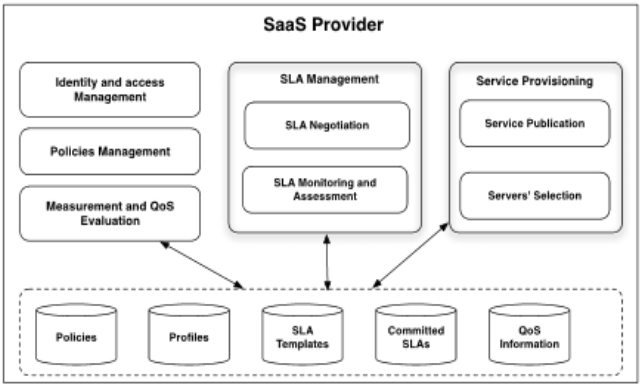

Figure 8. *Policy of SaaS cloud delivery*

Openstack Cloud Deployment

Study Case

OpenStack is extremely progressive through its powerful scalability to speak about specialized requirements of storage, networking and utility computing. The deployed cloud exists in on one logical server, Intel Xeon SR1670HVR. Server owns 16 GB as RAM and 1 TB of storage. The cloud includes 8 nodes such that 1 Controller node per system, 1 Compute node, 1 Network node, 1 block and 4 storage nodes. These nodes that are capable to implement the cloud are installed as VMs on the host CentOS 7 installed over nodes. Diverse Instances were effectively executed and established including Cirros, Debian, Ubuntu 18 server etc. on the Cloud.

The Hypervisor

The choice of hypervisor is extremely crucial for the performance of the cloud environment. QEMU is capable of shaping the OpenStack cloud in a VirtualBox environment. KVM does not own this ability because it needs material supports In order to be used, when using VirtualBox, support for virtualization in the physical machine BIOS was supported, then choice of the virtualization engine in the virtual machine's processor assets has been strengthened.

Executing instances

Firstly a SSH key shall be made so that instance might be accessible and flexible. Then an appropriate instances type must be chosen depending upon the exigencies. For this case (16GB RAM, 1TB storage) instance types are implemented. The execution of instances are accessible over the utility of Putty through the key-pair generated in the basic steps**.**

The Deployed Architecture

The installed cloud exist in on a particular server. Several nodes such as controller, compute and network are installed as virtual machines as presented on Fig.9

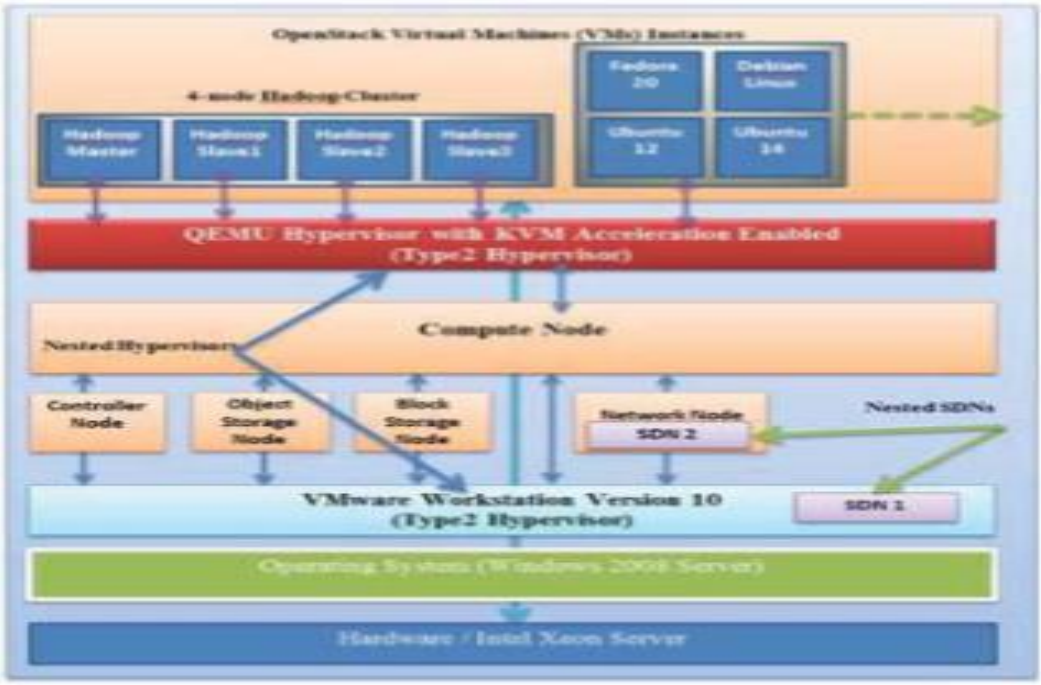

Figure 9. *Deployed Architecture of SaaS Cloud Service*

The Controller Node

It is essential topic of the cloud services and controls all other nodes of cloud. Elementary services offered through this node are the Dashboard, Image service and Identity service. Management portions of Compute and Network nodes are sustained here as well. Networking plug-in as presented in Fig.9. This offers service supports such as Database, message dealers or broker service and Network time protocol (NTP).

The Compute Node

It offers existing IT services. This node has the task of running the hypervisor. Instances are implemented on this node. This also administers the layer agents that run the tenant networks, the networking plug-in, and the execution of security sets.

The Network Node

This entity of OpenStack offers and monitors the set of networking services. It provides various layers 2 and layer 3 services like implementation of virtual network, NAT, DHCP and route. Internet connection of all instances is the task of this node.

Methodology

Linux Services

Linux service is basic software which executes in background mode to perform certain main and primary tasks in order to be utilized [22]. Several network service entities are accessible specifically NFS, TELNET, SSH etc. These tests are extended with rpcbind. The RPC in rpcbind is positioned for the remote method call. RPC is an inter-process communication which enables permits a computer program to trigger the process or subprogram in address space. It enables the host to make RPC calls on a server [24]. In order to evaluate rpcbind service within the Cloud, the package requires to be configured as a service. In most of Linux distribution does not contain default configurations.

Performance Metrics

Though a set of metrics are acquired from the experimentation but lapsed time is one of further prominent metric. Lapsed time is the deadline to accomplish an experience. It is the variance between start time and a completion time. Linux time command is used to size the lapsed timeThe similarity of information might be found from resource administrator web interface but Linux time command is ideal [14].Lapsed time can simply distinguish the performance, as the outcomes getting less lapsed time are reflected to obtain suitable dynamic and accessibility.

Vi. Comparison of the Cloud Platforms

We briefly define the results of the completed comparison of the perceived cloud platforms. Table 2 plainly indicates the variances between latest versions of the experimental open source cloud platforms.

Table 2. *Comparative study of Open Source IaaS solutions*

Supports Service Models	**OpenStack IaaS**	**Eucalyptus IaaS**	**Open Nebula IaaS**
Types of Cloud	Public Private Hybrid	Private Hybrid	Private Hybrid
Operating System	Linux	Linux	Linux
Packages Available	Redhat, CentOS Debian, Opensuse Ubuntu	Redhat CentOS	Redhat CentOS Fedora
Hypervisor	XenSerer, VMware, KVM,Xen, Hyperv,Lxc, QEMU, Uml,PowerM, BareMetal	,Xen, KVM, Vmware, XenServer	KVM, VMwae Xen, XenServer
VM OSs	Windows Linux	Windows Linux	Windows, Linux
Ease of Utilization resource Management Strategies	Yes Web Interfaces	Yes Web Interfaces	Yes Web Interfaces

Conclusion

OpenStack is intended to allow investigators and administrator systems to set up the IaaS infrastructure and compromises tools in order to create and manage underlying virtual machines in addition to existing resources. This work intents to clarify that the OpenStack cloud system has accomplished an essential niche in the design space of cloud computing by offering an easy deployment way on current resources, relaxed to perform in experimentation by being modular, and especially in open source form and offers powerful functionalities. With cloud computing technology and the increasing number of SaaS vendors, service clients will progressively face the challenge of discovery right SaaS vendors who could be capable to meet their functional and non-functional necessities. In this article, we presented the open source cloud OPENSTACK to develop the deployment of SaaS cloud services, followed by SLA. In our future work, we intend to outline the quality of service restrictions and the different illustrations used by SaaS vendors and to perform a prototype framework with some real situations for the provision of services based on SLA.

References

[1]Quality of Service and Service Level Agreements for Cloud Environments: Issues and Challenges [2] https://www.openstack.org/assets/survey/April-2016-User-SurveyReport.pdf

[3] P. Mell and T. Grance," The NIST Definition of Cloud Computing", Technical report, National Institute of Standards and Technology, 2009.
[4]http://americas.nttdata.com/blogs/cloud/2014/june/cloud-seriespart-3, last accessed on: 20/01/2017.
[5] Dr. H. Guruprasad, "Building Private Cloud using OpenStack", International Journal of Emerging Trends & Technology in Computer Science (IJETTCS),2014. [6] https://www.openstack.org/software
[7] Open source software for creating private and public clouds,https://www.openstack.org/
[8] Parakh, P., D G, N., Moin Mulla, M., & Baligar, V. P. (2018). SLA-awareVirtual Machine Scheduling in OpenStack-based Private Cloud. 2018 3rd International Conference on Computational Systems and information Technology for Sustainable Solutions (CSITSS). doi:10.1109/csitss.2018.8768760,https://scihub.bban.top/https://ieeexplore.ieee.org/document/8768760/ [9]https://www.answercart.com/blog/opensource-vssaas
[10]Soham Dutta, Introduction to OpenStack, https://medium.com/swlh/introduction-to-openstack-f750776e551a
[11]Vaquero LM, Rodero-Merino L, Morn D (2011) Locking the sky: a survey on IaaS cloud security
[12] Pang Xiong Wen, Li Dong, Quality Model for Evaluating SaaS Service, https://www.academia.edu/6483846/Quality_Model_for_Evaluating_SaaS_Service
[13]Flexera,HowtoDeliverSaaSUsingIaaS,https://www.flexera.com/blog/cloud/2013/09/how-to-deliversaasusing-iaas/
[14]ElarbiBadidi,A_Framework_for_SoftwareasaService_Selection,https://www.researchgate.net/publication/237082398 [15]https://opensourceforu.com/2018/11/installing-openstack-queens-on-a-centos-7-server/ [16] https://www.idginsiderpro.com/article/3262526/how-to-make-your-slas-meaningful-in-saascloudsubscription-agreements.html
[17] https://wiki.openstack.org/wiki/Packstack
[18]John Harauz. Data Security in the World of Cloud Computing[J].IEEE SECURITY &PRIVACY,2009,7(4)61-64| [19] S.Subashini. A survey on security issues in service delivery models of cloud computing[J].Journal of Networkand Computer Applications,2011,34:1-11
[20] Cloud Computing Use Case Discussion Group. Cloud Computing Use Cases White Paper[EB/OL]. 2010.02, http://cloudusecases.org/, retrived 2011-08-15
[21]HadooponOpenStack,http://hortonworks.com/labs/openstack
[22]Michael G. Noll, "Benchmarking and Stress Testing an HadoopClusterwithTeraSort,TestDFSIO&Co",http://www.michaelnoll.com/blog/2011/04/09/benchmarking-and-stress-testing-an-hadoopcluster-with-terasort-testdfsio-nnbench-mrbench/ [23]https://computingforgeeks.com/deployopenstack-cloud-on-centos-rhel-with-packstack-rdo/
[24]LinuxServices,Available[online]:http://www.linux.com/news/enterprise/systemsmanagement/8116-an-introduction-to-servicesrunlevels-and-rcd-scripts